



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/623,023	07/18/2003	Robert J. Madril JR.	200208758-1	4467
22879 7590 11/23/2007 HEWLETT PACKARD COMPANY P O BOX 272400, 3404 E. HARMONY ROAD INTELLECTUAL PROPERTY ADMINISTRATION FORT COLLINS, CO 80527-2400			EXAMINER DICKERSON, CHAD S	
			ART UNIT 2625	PAPER NUMBER
			MAIL DATE 11/23/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/623,023	MADRIL ET AL.	
	Examiner	Art Unit	
	Chad Dickerson	2625	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
 - If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
 - Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 13 September 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-36 is/are pending in the application.
- 4a) Of the above claim(s) 6, 11-13, 20-22, 25, 26 and 31-33 is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-5, 8-10, 14-19, 23, 24, 27-30 and 34-36 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 7/18/2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Response to Arguments

1. Applicant's arguments, see page 13, filed 9/13/2007, with respect to 112 second paragraph rejections have been fully considered and are persuasive. The 112 second paragraph rejections of claims 8, 10, 16 and 24 have been withdrawn.
2. Applicant's arguments with respect to claims 1-36 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1-5, 8-10, 14, 19, 27-30, 34 and 35 rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno '549 (US Pub No 2003/0135549) in view of Anderson '624 (US Pub No 2003/0067624) and Kemp '160 (US Pub No 2002/0078160).

Re claim 1: Kuno et al discloses print system including application server comprising processor-executable instructions configured for:

generating a content type that identifies output that is compatible with a printer identified by the printer identification (i.e. the server then makes PDL data using the user's print request that is suitable for the registered printer designated by the user

Art Unit: 2625

when informing the server (102) about the printer type. The content type that identifies or represents an output that is compatible to the printer is the PDL data that is generated by the server (102); see fig. 1, 5 and 6; paragraphs [0036] and [0037]); and

However, Kuno '549 fails to teach receiving from a mobile computing device, print instructions and an email having an attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed; transferring the attachment, the finishing options, and the content type to a translator engine; translating the attachment into print-ready data using the finishing options and a printer driver identified through information included within the content type, and forwarding, the print-ready data to the printer identified by the print identification.

However, this is well known in the art as evidenced by Anderson '624. Anderson '624 discloses receiving from a mobile computing device (i.e. the element 183, which includes a PDA, pager, cell phone or laptop; see fig. 1; paragraph [0026]), print instructions and an email having mail attachment (i.e. the mobile device sends an email with an attachment of documents. The email with the attachment are sent, reformatted, to the MPC (106) with special printer requirements that the user desires to be reflected in the output of the attachment; see paragraphs [0035]-[0044]),

the print instructions including a printer identification and document options indicating how the attachment should be printed (i.e. in the information determining the printing of the email attachment, the mobile device sends a printer connector identifier (179) that is associated with the MPC (106), which informs the MPC server (143) which

Art Unit: 2625

is the destination device to perform the printing of the rendered email attachment. Also, other special printing requirements and settings for the printer that indicate how the user want the attachment to be printed is also included in the information; see paragraphs [0035]-[0044]);

transferring the attachment, the options, and the content type to a translator engine (i.e. the mobile device transfers the email attachment, the special printing settings, which is considered as the options, or requirements and the address of the mobile print server (143) that is going to render the email attachment into the rendered version that will be printed by the printer. The domain associated with the MPC service (143) is considered to be the content type since this recognizes the appropriate translating device to render the attachment into data that can be printed by the system's printer; see paragraphs [0034]-[0044]);

translating the attachment into print-ready data using the options (i.e. the MPC server (143) translates the email's attachment using the different printer settings and other user preferences into data that is ready to be printed by the printer (169) used in the system; see paragraphs [0035]-[0044]) and

a printer driver identified through information included within the content type (i.e. the MPC server (143) is identified using the destination address, which is considered as the content type. Other types of information are also included in the email sent by the user to the service (103). All the information regarding the MPC service (143) and the MPC (106) are all considered as content type information; see paragraphs [0035]-[0044]), and

forwarding, the print-ready data to the printer identified by the print identification (i.e. in the system, the email also includes the destination device that will output, or forward, the rendered version of the attachment of the email. This information is sent to the printer that is identified by the printer connector identifier; see paragraphs [0035]-[0044]).

Therefore, in view of Anderson '624, it would have been obvious to one of ordinary skill at the time the invention was made to have the features of receiving from a mobile computing device, print instructions and an email having ml attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed; transferring the attachment, the finishing options, and the content type to a translator engine; translating the attachment into print-ready data using the finishing options and a printer driver identified through information included within the content type, and forwarding, the print-ready data to the printer identified by the print identification in order to receive an attachment recognizable by the printer in the mobile printer connector and printing the attachment (as stated in Anderson '624 paragraphs [0007]-[0010]).

However, Kuno '549 in view of Anderson '624 fails to teach finishing options.

However, this is well known in the art as evidenced by Kemp '160. Kemp '160 discloses finishing options (i.e. in Kemp '160, the user sends finishing request with the printing request to the service provider (2) from the computer workstation. This feature combined with the feature of Anderson '624 to provide options to print the document in

Art Unit: 2625

a special manner by the printer performs the above feature; see fig. 1; paragraph [0041]).

Therefore, in view of Kemp '160, it would have been obvious to one of ordinary skill at the time the invention was made to have finishing options in order to have the user be able to submit a printer request involving a finishing process (as stated in Kemp '160 paragraph [0041]).

Re claim 2: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a processor-readable medium, further comprising processor-executable instructions configured for:

determining a printer driver that corresponds with the printer identification (i.e. when the user registers a printer that he or she wants to use and then informs the server of the printer type, this is an example of informing the server of the printer identification. Then the server (102) attempts to designate the printer name to locators of printer drivers and determine if there is a printer driver that corresponds to the printer type, or printer identification. Then the server installs the printer driver suitable to the designated printer; see figs. 1, 5 and 6; paragraphs [0036] and [0037]); and

including in the content type, information identifying the printer driver (i.e. the information relating to the printer type is used to locate the printer drivers suitable for the printers identified. Therefore, this information is considered to identify printer drivers in the system; see figs. 1, 5 and 6; paragraphs [0036] and [0037]).

Re claim 3: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al a processor-readable medium, further comprising processor-executable instructions configured for:

determining if the printer driver has been installed (i.e. in the system, the invention has the ability to allow the user to obtain a printer driver if it is not currently on the server. In order to know that a driver needs to be installed on the server, the determination that a printer driver is or is not on the server has to be made. Therefore, it is clear that the above feature is performed; see fig. 5 and 6; paragraphs [0036] and [0037]); and

automatically installing the printer driver if the printer driver has not already been installed (i.e. whenever a print request is received by the server (102), an automatic installation of a printer driver is performed. When the server realizes that it does not have the appropriate printer driver to perform a print job, it can obtain a printer driver from a URL or some other source and install the printer driver; see fig. 1 and 5; paragraphs [0029] and [0036]).

Re claim 4: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a processor-readable medium, wherein the determining comprises:

accessing a lookup table (i.e. the server (102) has a table that relates the printer names to the locators of the printer drivers. This table is accessed by the server in order to acquire a suitable printer driver to the printer type; see fig. 5 and 6; paragraphs [0036] and [0037]); and

comparing the printer identification with printer driver entries in the lookup table (i.e. the printer names or types are related, or compared, to the locators of the printer drivers. Then the server installs the printer driver that is most suitable to the designated printer; see fig. 5 and 6; paragraphs [0036] and [0037]).

Re claim 5: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a processor-readable medium, wherein the translating the attachment into print-ready data further comprises:

sending from the translator engine to a driver management service, a request to map the printer driver to a printer icon in a printer folder (i.e. in the system, the client computer is used to store a program that utilizes a program to make print data and to utilize a printer driver. The client computer can be considered as the translator engine since it utilizes a program to make the document data into print data that can be interpreted by a print controller. The client computer sends a request to a server to obtain an application program; see paragraphs [0031]-[0037]), a request to map the printer driver to a printer icon in a printer folder (i.e. the user registers a printer that he or she desires to use. Then the printer information is sent to the application server and the

Art Unit: 2625

server will relate the printer name, which is a representation of the printer and also considered analogous to a printer icon, to the printer drivers that can be used to work with the respective printer. The relation between the printer drivers and printer names are stored in a table. The table is considered analogous to the printer folder because both store information pertaining to the printer name; see paragraphs [0031]-[0037]);

However, Kuno fails to teach in response to the request, at the translating engine a printer icon name; and implementing a print command at the translator engine using the printer icon name.

However, this is well known in the art as evidenced by Anderson '624. Anderson '624 discloses in response to the request, receiving at the translating engine a printer icon name (i.e. in the system of Anderson '624, the MPC service (143) performs the feature of translating an email's attachment into rendered form that can be printed by the printer (169). When an email is sent to the mobile print server (103) in regards to a print request, the MPC service (143) receives a printer MPC identifier (179) that represents a printer to be used as a destination device to print out a document. This performs the feature of in response to a print request, the MPC service (143) receives a printer or MPC identifier, considered as a printer icon since it represents a desired printer, identifying a printer to use for printing; see paragraphs [0028]-[00045]); and

implementing a print command at the translator engine using the printer icon name (i.e. at the MPC service (143), a print command is implemented using the MPC identifier (179) of the destination MPC (106) regarding the printer to be used for printing. Using this identifier in the email and executing a printing operation using the identifier is

Art Unit: 2625

an example of using an identifier that represents a printer, analogous to a printer icon, to be used by the MPC service (143) in order to know that destination device the rendered document should be output to for printing; see paragraphs [0028]-[00045]).

Therefore, in view of Anderson '624, it would have been obvious to one of ordinary skill at the time the invention was made to have the features of in response to the request, at the translating engine a printer icon name; and implementing a print command at the translator engine using the printer icon name in order to inform a server the destination device for the rendered version of an attachment of an email (as stated in Anderson '624 paragraphs [0036]-[0040]).

Re claim 8: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a processor-readable medium, wherein the content type comprises information selected from the group comprising:

information identifying a printer driver (i.e. the server has information relating to the content type that identifies the printer driver stored in the hard disk drive (116); see fig. 2, 5 and 6; paragraphs [0031], [0033] and [0036]-[0039]);

information identifying capabilities of a printer driver (i.e. with the server (102) having the different drivers stored on the hard disk drive (116) related to the different client devices, this relates to and identifies information about the capability of the printer drivers present on the server; see fig. 2, 5 and 6; paragraphs [0031], [0033] and [0036]-[0039]);

information describing output that is compatible with the printer (i.e. when the server is looking for a suitable printer driver for the printer name, it looks at the table in the server and activates a driver. This driver activation is present on the monitor (118) on the server (102). Then the information that is outputted to the client PC is information that describes an output that is compatible with the printer; see fig. 2, 5 and 6; paragraphs [0031], [0033] and [0036]-[0039]);

information describing a compatible printer page description language (i.e. the print data outputted is used to be described with a page description language that is appropriate to the printer in which the print image data is outputted to; see fig. 2, 5 and 6; paragraphs [0031]-[0033] and [0036]-[0039]); and

information identifying a compatible printer (i.e. when the user sends to the server information designating a printer type that is used by the server to apply a certain printer driver, then information identifying a compatible printer is received from the client PC to the server; see fig. 2, 5 and 6; paragraphs [0031]-[0033] and [0036]-[0039]).

Re claim 9: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a processor-readable medium, wherein the receiving a printer identification includes receiving a request to map the printer identification to a content type (i.e. when the user requests a print, the server receives the request to link the printer icon name, that represents a printer driver, to the content type of the printer designated for printing, or printer type designated which is analogous to the printer

Art Unit: 2625

identification. The system clearly performs the above feature; see figs. 5 and 6; paragraphs [0036] and [0037]).

Re claim 10: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a processor-readable medium wherein the printer identification is selected from the group comprising:

a text string specifying a printer model (i.e. the hard disk drive (116) stores the model names of printer that are used by the server to relate printer names to the printer driver; see fig. 2, 5 and 6; paragraphs [0031], [0036] and [0037]);

a text string specifying the printer driver (i.e. the hard disk drive (116) stores the names of printer drivers that are accessed by the server. The printer driver names specify the printer drivers in relation to the printer names; see fig. 2, 5 and 6; paragraphs [0031], [0036] and [0037]); and

a UNC (Universal Naming Convention) path (i.e. the registration information of client computers are stored in folders, or as directories, for access by the server. These folders or directories are analogous to a UNC since the directories represent a path to the registration information; see fig. 1 and 2; paragraphs [0031]) identifying a location of the printer driver (i.e. the system uses a URL, which is analogous to a UNC, to locate a printer driver that may be needed by the server; see fig. 2, 5 and 6; paragraphs [0029]-[0037]).

Re claim 14: Kuno et al discloses a print system application server comprising processor-executable instructions configured for:

generating a content type that identifies output that is compatible with a printer identified by the printer identification (i.e. the server then makes PDL data using the user's print request that is suitable for the registered printer designated by the user when informing the server (102) about the printer type. The content type that identifies or represents an output that is compatible to the printer is the PDL data that is generated by the server (102); see fig. 1, 5 and 6; paragraphs [0036] and [0037]); and

receiving a request to map a printer icon name to a content type (i.e. when the user requests a print, the server receives the request to link the printer icon name, that represents a printer driver, to the content type of the printer designated for printing, or printer type. The system clearly performs the above feature; see figs. 5 and 6; paragraphs [0036] and [0037]);

mapping a printer icon name to the content type (i.e. the printer name is related, or mapped, to the printer driver that will output a specific content type to the local printer on the client PC. Therefore, the printer name, which represents a printer driver, is mapped or associated with a content type that will be outputted to the local printer; see figs. 5 and 6; paragraphs [0036] and [0037]);

determining if a printer driver associated with the printer icon name is available (i.e. when the user registers a printer that he or she wants to use and then informs the server of the printer type, this is an example of informing the server of the printer identification. Then the server (102) attempts to designate the printer name to locators

Art Unit: 2625

of printer drivers and determine if there is a printer driver that corresponds to the printer type, or printer identification, is available to use. Then the server installs the printer driver suitable to the designated printer; see figs. 1, 5 and 6; paragraphs [0036] and [0037]); and

returning the printer icon name if the printer driver is available (i.e. when the CPU (110) checks the hard disk drive to see if a certain printer driver is available, the hard disk drive (116) returns to the CPU (110) the printer driver locator, analogous to a printer icon name, related to a certain printer. At this point, the printer driver may not necessarily need to be reinstalled. Also, when the server (102) checks a URL for a certain printer driver, the URL returns a printer driver in relation to the URL being searched; see figs. 1, 5 and 6; paragraphs [0036] and [0037]).

However, Kuno '549 fails to teach receiving from a mobile computing device, print instructions and an email having an attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed.

However, this is well known in the art as evidenced by Anderson '624. Anderson '624 discloses receiving from a mobile computing device (i.e. the element 183, which includes a PDA, pager, cell phone or laptop; see fig. 1; paragraph [0026]), print instructions and an email having mail attachment (i.e. the mobile device sends an email with an attachment of documents. The email with the attachment are sent, reformatted, to the MPC (106) with special printer requirements that the user desires to be reflected in the output of the attachment; see paragraphs [0035]-[0044]),

the print instructions including a printer identification and document options indicating how the attachment should be printed (i.e. in the information determining the printing of the email attachment, the mobile device sends a printer connector identifier (179) that is associated with the MPC (106), which informs the MPC server (143) which is the destination device to perform the printing of the rendered email attachment. Also, other special printing requirements and settings for the printer that indicates how the user wants the attachment to be printed is also included in the information; see paragraphs [0035]-[0044]).

Therefore, in view of Anderson '624, it would have been obvious to one of ordinary skill at the time the invention was made to have the features of receiving from a mobile computing device, print instructions and an email having ml attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed in order to receive an attachment recognizable by the printer in the mobile printer connector and printing the attachment (as stated in Anderson '624 paragraphs [0007]-[0010]).

However, Kuno '549 in view of Anderson '624 fails to teach finishing options.

However, this is well known in the art as evidenced by Kemp '160. Kemp '160 discloses finishing options (i.e. in Kemp '160, the user sends finishing request with the printing request to the service provider (2) from the computer workstation. This feature combined with the feature of Anderson '624 to provide options to print the document in a special manner by the printer performs the above feature; see fig. 1; paragraph [0041]).

Therefore, in view of Kemp '160, it would have been obvious to one of ordinary skill at the time the invention was made to have finishing options in order to have the user be able to submit a printer request involving a finishing process (as stated in Kemp '160 paragraph [0041]).

Re claim 19: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a processor-readable medium, further comprising processor-executable instructions configured for:

determining if the printer driver has been installed (i.e. in the system, the invention has the ability to allow the user to obtain a printer driver if it is not currently on the server. In order to know that a driver needs to be installed on the server, the determination that a printer driver is or is not on the server has to be made. Therefore, it is clear that the above feature is performed; see fig. 5 and 6; paragraphs [0036] and [0037]); and

automatically installing the printer driver if the printer driver has not already been installed (i.e. whenever a print request is received by the server (102), an automatic installation of a printer driver is performed. When the server realizes that it does not have the appropriate printer driver to perform a print job, it can obtain a printer driver from a URL or some other source and install the printer driver; see fig. 1 and 5; paragraphs [0029] and [0036]).

Art Unit: 2625

Re claim 27: Kuno et al discloses a print system including application server comprising:

a content transformation harness configured to receive a document and a printer identification (i.e. the user edits information related to the desired document to be outputted and it informs the server of the printer type that will be used for local printing. The printer type can be considered as the printer identification. Also, the client PC can also send print data, which is considered to be a document, can be sent to the server to be processed into PDL data. Although a content transformation harness is not specifically disclosed, the feature is still performed; see figs. 5 and 6; paragraphs [0022]-[0025], [0031]-[0036] and [0037]);

a driver management service (102) configured to determine a printer driver corresponding with the printer identification (i.e. when the user registers a printer that he or she wants to use and then informs the server of the printer type, this is an example of informing the server of the printer identification. Then the server (102) attempts to designate the printer name to locators of printer drivers and determine if there is a printer driver that corresponds to the printer type, or printer identification. Then the server installs the printer driver suitable to the designated printer; see figs. 1, 5 and 6; paragraphs [0036] and [0037]) and to install the printer driver if the printer driver is not already installed (i.e. whenever a print request is received by the server (102), an automatic installation of a printer driver is performed. When the server realizes that the it does not have the appropriate printer driver to perform a print job, it can obtain a printer driver from a URL or some other source and install the printer driver; see fig. 1 and 5; paragraphs [0029] and [0036]); and

Art Unit: 2625

a translator engine configured to transform the document into print-ready data using the printer driver (i.e. the printer drivers on the hard disk drive (116) are able to transform the print data into PDL or print-ready data; see fig. 5 and 6; paragraphs [0031]-[0037]).

However, Kuno '549 fails to teach receiving from a mobile computing device, print instructions and an email having an attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed, the MEP server further to separate the document from the email, a content transformation harness configured to receive a document and a printer identification from the MEP server and wherein the MEP server receives the print-ready data and forwards the print-ready data to an appropriate printer identified by the printer identification.

However, this is well known in the art as evidenced by Anderson '624. Anderson '624 discloses receiving from a mobile computing device (i.e. the element 183, which includes a PDA, pager, cell phone or laptop; see fig. 1; paragraph [0026]), print instructions and an email having mail attachment (i.e. the mobile device sends an email with an attachment of documents. The email with the attachment are sent, reformatted, to the MPC (106) with special printer requirements that the user desires to be reflected in the output of the attachment; see paragraphs [0035]-[0044]),

the print instructions including a printer identification and document options indicating how the attachment should be printed (i.e. in the information determining the printing of the email attachment, the mobile device sends a printer connector identifier

Art Unit: 2625

(179) that is associated with the MPC (106), which informs the MPC server (143) which is the destination device to perform the printing of the rendered email attachment. Also, other special printing requirements and settings for the printer that indicates how the user wants the attachment to be printed is also included in the information; see paragraphs [0035]-[0044]),

the MEP server further to separate the document from the email (i.e. in Anderson '624, the email attachment, which may consist of a document or documents, is separated from the email and the rendering application is applied to the attachment(s); see paragraphs [0035]-[0045]);

a content transformation harness configured to receive a document and printer identification from the MEP server (i.e. in the system, when the proxy server (186) can be used to send information regarding the attached document and the identification of the printer to where a document should be printed. In this scenario, the Mobile print server (103) can be used as a content transformation harness since it is used to transform the document information into a rendering form compatible for printing by a respective printer; see fig. 1; paragraphs [0028]-[0045]); and

wherein the MEP server receives the print-ready data and forwards the print-ready data to an appropriate printer identified by the printer identification (i.e. in the system, the MPC service (143) performs the rendering of the document and this document is sent to the MPC (106). This is scenario, the MPC (106) acts as a server since it receives the rendered, or print-ready data, and this data is sent to a previously

identified printer. The printer is identified by printer identification sent from the printer (169) to the MPC (106); see paragraphs [0028]-[0045]).

Therefore, in view of Anderson '624, it would have been obvious to one of ordinary skill at the time the invention was made to have the features of receiving from a mobile computing device, print instructions and an email having an attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed, the MEP server further to separate the document from the email; a content transformation harness configured to receive a document and a printer identification from the MEP server and wherein the MEP server receives the print-ready data and forwards the print-ready data to an appropriate printer identified by the printer identification in order to receive an attachment recognizable by the printer in the mobile printer connector and printing the attachment (as stated in Anderson '624 paragraphs [0007]-[0010]).

However, Kuno '549 in view of Anderson '624 fails to teach finishing options.

However, this is well known in the art as evidenced by Kemp '160. Kemp '160 discloses finishing options (i.e. in Kemp '160, the user sends finishing request with the printing request to the service provider (2) from the computer workstation. This feature combined with the feature of Anderson '624 to provide options to print the document in a special manner by the printer performs the above feature; see fig. 1; paragraph [0041]).

Therefore, in view of Kemp '160, it would have been obvious to one of ordinary skill at the time the invention was made to have finishing options in order to have the

Art Unit: 2625

user be able to submit a printer request involving a finishing process (as stated in Kemp '160 paragraph [0041]).

Re claim 28: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a server, further comprising:

a content type, generated by the driver management service and containing information indicating the printer driver (i.e. the server (102) generates a content type that contains information that indicates the printer driver on the server (102) ; see fig. 2, 5 and 6; paragraphs [0031]-[0037]) and capabilities of the printer driver (i.e. with the hard disk drive (116) storing different printer drivers that are in relation to different client PCs using the system, this is information that shows the different capabilities that the printer drivers can perform on the server; see fig. 2, 5 and 6; paragraphs [0031]-[0037]).

Re claim 29: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a server, further comprising:

a driver look-up table containing printer driver entries and associated printer identifications (i.e. the server (102) has a table that relates the printer names to the locators of the printer drivers; see fig. 5 and 6; paragraphs [0036] and [0037]), the driver look-up table used by the driver management service to determine a printer driver (i.e.

Art Unit: 2625

this table is accessed by the server in order to acquire a suitable printer driver to the printer type; see fig. 5 and 6; paragraphs [0036] and [0037]).

Re claim 30: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al discloses a server, further comprising:

uninstalled driver files used by the driver management service to install the printer driver if the printer driver is not already installed (i.e. whenever a print request is received by the server (102), an automatic installation of a printer driver is performed.

When the server realizes that it does not have the appropriate printer driver to perform a print job, it can obtain a printer driver from a URL or some other source and install the printer driver. The printer drivers that are present on the client device or the URL are considered to be uninstalled drivers since these drivers are not on the server. The server then acquires these drivers that are uninstalled and installs them on the server. This only occurs if this process is necessary; see fig. 1, 5 and 6; paragraphs [0029] and [0036]-[0045]).

Re claim 34: Kuno et al discloses a server comprising:

means for generating a content type that identifies output that is compatible with a printer identified by the printer identification (i.e. the server then makes PDL data using the user's print request that is suitable for the registered printer designated by the user when informing the server (102) about the printer type. The content type that

identifies or represents an output that is compatible to the printer is the PDL data that is generated by the server (102); see fig. 1, 5 and 6; paragraphs [0036] and [0037]).

However, Kuno '549 fails to teach means for receiving from a mobile computing device, print instructions and an email having an attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed, means for separating the document from the email; means for transferring the attachment, the finishing options, and the content type to a translator engine; means for translating the attachment into print-ready data using the finishing options and a printer driver identified through information included within the content type, and means for forwarding, the print-ready data to the printer identified by the print identification.

However, this is well known in the art as evidenced by Anderson '624. Anderson '624 discloses means for receiving from a mobile computing device (i.e. the element 183, which includes a PDA, pager, cell phone or laptop; see fig. 1; paragraph [0026]), print instructions and an email having mail attachment (i.e. the mobile device sends an email with an attachment of documents. The email with the attachment are sent, reformatted, to the MPC (106) with special printer requirements that the user desires to be reflected in the output of the attachment; see paragraphs [0035]-[0044]),

the print instructions including a printer identification and document options indicating how the attachment should be printed (i.e. in the information determining the printing of the email attachment, the mobile device sends a printer connector identifier (179) that is associated with the MPC (106), which informs the MPC server (143) which

Art Unit: 2625

is the destination device to perform the printing of the rendered email attachment. Also, other special printing requirements and settings for the printer that indicates how the user wants the attachment to be printed is also included in the information; see paragraphs [0035]-[0044]),

means for separating the document from the email (i.e. in Anderson '624, the email attachment, which may consist of a document or documents, is separated from the email and the rendering application is applied to the attachment(s); see paragraphs [0035]-[0045]);

means for transferring the attachment, the options, and the content type to a translator engine (i.e. the mobile device transfers the email attachment, the special printing settings, which is considered as the options, or requirements and the address of the mobile print server (143) that is going to render the email attachment into the rendered version that will be printed by the printer. The domain associated with the MPC service (143) is considered to be the content type since this recognizes the appropriate translating device to render the attachment into data that can be printed by the system's printer; see paragraphs [0034]-[0044]);

means for translating the attachment into print-ready data using the options (i.e. the MPC server (143) translates the email's attachment using the different printer settings and other user preferences into data that is ready to be printed by the printer (169) used in the system; see paragraphs [0035]-[0044]) and

a printer driver identified through information included within the content type (i.e. the MPC server (143) is identified using the destination address, which is considered as

Art Unit: 2625

the content type. Other types of information are also included in the email sent by the user to the service (103). All the information regarding the MPC service (143) and the MPC (106) are all considered as content type information; see paragraphs [0035]-[0044]), and

means for forwarding, the print-ready data to the printer identified by the print identification (i.e. in the system, the email also includes the destination device that will output, or forward, the rendered version of the attachment of the email. This information is sent to the printer that is identified by the printer connector identifier from the mobile print server (103); see paragraphs [0035]-[0044]).

Therefore, in view of Anderson '624, it would have been obvious to one of ordinary skill at the time the invention was made to have the means for receiving from a mobile computing device, print instructions and an email having an attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed, means for separating the document from the email; means for transferring the attachment, the finishing options, and the content type to a translator engine; means for translating the attachment into print-ready data using the finishing options and a printer driver identified through information included within the content type, and means for forwarding, the print-ready data to the printer identified by the print identification in order to receive an attachment recognizable by the printer in the mobile printer connector and printing the attachment (as stated in Anderson '624 paragraphs [0007]-[0010]).

However, Kuno '549 in view of Anderson '624 fails to teach finishing options.

However, this is well known in the art as evidenced by Kemp '160. Kemp '160 discloses finishing options (i.e. in Kemp '160, the user sends finishing request with the printing request to the service provider (2) from the computer workstation. This feature combined with the feature of Anderson '624 to provide options to print the document in a special manner by the printer performs the above feature; see fig. 1; paragraph [0041]).

Therefore, in view of Kemp '160, it would have been obvious to one of ordinary skill at the time the invention was made to have finishing options in order to have the user be able to submit a printer request involving a finishing process (as stated in Kemp '160 paragraph [0041]).

Re claim 35: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al teaches a server, further comprising:

means for receiving a request to map a printer icon name to a content type (i.e. when the user requests a print, the server receives the request to link the printer icon name, that represents a printer driver, to the content type of the printer designated for printing, or printer type. The system clearly performs the above feature; see figs. 5 and 6; paragraphs [0036] and [0037]);

means for mapping a printer icon name to the content type (i.e. the printer name is related, or mapped, to the printer driver that will output a specific content type to the local printer on the client PC. Therefore, the printer name, which represents a printer

Art Unit: 2625

driver, is mapped or associated with a content type that will be outputted to the local printer; see figs. 5 and 6; paragraphs [0036] and [0037]);

means for determining if a printer driver associated with the printer icon name is available (i.e. when the user registers a printer that he or she wants to use and then informs the server of the printer type, this is an example of informing the server of the printer identification. Then the server (102) attempts to designate the printer name to locators of printer drivers and determine if there is a printer driver that corresponds to the printer type, or printer identification, is available to use. Then the server installs the printer driver suitable to the designated printer; see figs. 1, 5 and 6; paragraphs [0036] and [0037]); and

means for returning the printer icon name if the printer driver is available (i.e. when the CPU (110) checks the hard disk drive to see if a certain printer driver is available, the hard disk drive (116) returns to the CPU (110) the printer driver locator, analogous to a printer icon name, related to a certain printer. At this point, the printer driver may not necessarily need to be reinstalled. Also, when the server (102) checks a URL for a certain printer driver, the URL returns a printer driver in relation to the URL being searched; see figs. 1, 5 and 6; paragraphs [0036] and [0037]).

5. Claim 7 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno '549, as modified by Anderson '624 and Kemp '160, and further in view of Gunji (US Pub No 2003/0065755).

Art Unit: 2625

Re claim 7: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

However, Kuno et al in view of Anderson '624 and Kemp '160 fails to teach a processor-readable medium, wherein the content type is a structured XML (Extensible Markup Language) document.

However, this is well known in the art as evidenced by Gunji. Gunji discloses the content type is a structured XML (Extensible Markup Language) document (i.e. the invention has a method where an XML type is used to describe attribute information that specifies the attributes of a driver provided on a server; see paragraphs [0012]-[0016]).

Therefore, in view of Gunji, it would have been obvious to one of ordinary skill at the time the invention was made to have a content type structured as an XML (Extensible Markup Language) document in order to analyze attribute information of a driver in a certain language (as stated in Gunji paragraph [0013]).

6. Claim 15 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno et al, as modified by Anderson '624 and Kemp '160, and further in view of Liang '297 (US Pat No 7212297).

Re claim 15: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

However, Kuno et al in view of Anderson '624 and Kemp '160 fails to teach a processor-readable medium, further comprising processor-executable instructions configured for returning an error message if the printer driver is not available.

However, this is well known in the art as evidenced by Liang '297. Liang '297 discloses returning an error message if the printer driver is not available (i.e. if a conventional printer driver is not available, an error message is generated; see fig. 8 and 9; col. 7, lines 4-59).

Therefore, in view of Liang '297, it would have been obvious to one of ordinary skill at the time the invention was made to return an error message if the printer driver is not available incorporated in the device of Kuno et al in order to generate an error message when a printer driver is not available (as stated in Liang '297 col. 7, lines 4-59).

7. Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno et al, as modified by Anderson '624, Kemp '160 and Liang '297, and further in view of Chang et al (US Pub No 2002/0059415).

Re claim 16: The teachings of Kuno et al in view of Anderson '624, Kemp '160 and further in view of Liang '297 are disclosed above.

Liang '297 teaches a processor-readable medium, wherein the error message is selected from the group comprising:

a message indicating that the printer driver is unavailable (i.e. if a conventional printer driver is not available, an error message is generated after the determination of a printer driver is not available. This message indicates that a printer driver is not available; see fig. 8 and 9; col. 7, lines 4-59); and

a message indicating that the printer driver is temporarily unavailable (i.e. if a conventional printer driver is not available, an error message is generated after the determination of a printer driver is not available. This message indicates that a printer driver is not available. Whether temporarily unavailable or unavailable, under the same condition, the printer driver is unavailable; see fig. 8 and 9; col. 7, lines 4-59).

However, Kuno et al in view of Anderson '624 and Kemp '160, and further in view of Liang '297 fails to teach that the request should be attempted later.

However, this is well known in the art as evidenced by Chang et al. Chang et al discloses that the request should be attempted later (i.e. the system notifies the user to try again or it notifies that user at a later time for the use of the printer, through the printer driver, when the actual requested service is available; see fig. 5; paragraph [0148]).

Therefore, in view of Chang et al, it would have been obvious to one of ordinary skill at the time the invention was made to have a message indicating that the request should be attempted later in order to notify the user when services become available (as stated in Chang et al paragraph [0148]).

8. Claims 17 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno '549, as modified by Anderson '624 and Kemp '160, and further in view of Terrill et al (US Pub No 2002/0188646).

Re claim 17: The teachings of Kuno '549 in view of Anderson '624 and Kemp '160 are disclosed above.

Kuno et al teaches determining that the printer driver is available and determining that the printer driver is not available (i.e. in the system, the invention has the ability to allow the user to obtain a printer driver if it is not currently on the server. In order to know that a driver needs to be installed on the server, the determination that a printer driver is or is not on the server has to be made. Therefore, it is clear that the system determines if a printer driver is or is not available. If the printer driver is available, then it can be immediately used and installation of a printer driver is not necessary, which is stated in paragraph [0037]. If it is determined that the printer driver is not available, the server obtains a printer driver suitable for outputting the content type; see fig. 5 and 6; paragraphs [0036] and [0037]).

However, Kuno et al in view of Anderson '624 and Kemp '160 fails to teach comparing a threshold number of print requests to a current number of print requests, if the current number is less than the threshold number and if the threshold number does not exceed the current number.

However, this is well known in the art as evidenced by Terrill et al. Terrill et al discloses

comparing a threshold number of print requests to a current number of print requests (i.e. the system discloses the threshold of a number of print jobs or print jobs completed. The system compares the pre or post print information and the print job

Art Unit: 2625

completion data to constraints in the system. If constraints in the system are reached, data is transferred to a management server; see paragraphs [0011], [0038] and [0054]);

if the current number is less than the threshold number (i.e. when comparing the print job completion information, post and pre print information with the threshold, the information is compared to see if the threshold is reached. If the threshold is not reached, it represents the information being compared to the threshold is lower than the threshold; see paragraphs [0011], [0038] and [0054]); and

if the threshold number does not exceed the current number (i.e. when the print job information attains the threshold of the system, then information from the data store is transferred to the management server. This is an example of the threshold not exceeding the current number or the print job number, but being equal to that number; see paragraphs [0011], [0038] and [0054]).

Therefore, in view of Terrill et al, it would have been obvious to one of ordinary skill at the time the invention was made to compare a threshold number of print requests to a current number of print requests, to judge if the current number is less than the threshold number or if the threshold number does not exceed the current number in order to judge to realization of the threshold in the system (as stated in Terrill et al paragraph [0011]).

Re claim 18: The teachings of Kuno et al in view of Anderson '624 and Kemp '160 and further in view of Terrill et al are disclosed above.

Kuno et al teaches a processor-readable medium, wherein the comparing comprises:

accessing a look-up table (i.e. the server (102) has a table that relates the printer names to the locators of the printer drivers. This table is accessed by the server in order to acquire a suitable printer driver to the printer type; see fig. 5 and 6; paragraphs [0036] and [0037]); and

determining the current number according to active print requests being processed by the printer driver (i.e. on the server (102), the hard disk drive (116) stores the current number of active printer drivers corresponding to the print request given from the client. With this information, the server (102) is able to determine the current number of print jobs because of the print requests that are correlated to the active printer drivers in the server that is processing the print requests; see fig. 1, 2, 5 and 6; see paragraphs [0029]-[0033]).

However, Kuno et al in view of Anderson '624 and Kemp '160 fails to teach a processor-readable medium, wherein the comparing comprises: accessing the threshold number from a look-up table.

However, this is well known in the art as evidenced by Terrill et al. Terrill et al discloses the comparing comprises:

accessing the threshold number from a look-up table (i.e. in the conventional in Terrill et al, job information is listed in a job table. In the system, the thresholds are contained in the data transfer module (214), data store (212) or another location. Although the system does not specifically state accessing a threshold number from a

Art Unit: 2625

look-up table, the threshold number is accessed from a various amounts of locations. If the threshold was on the job information collection and correlation module which has the job table, it may perform the feature of being accessed from a look-up table; see paragraphs [0004], [0037] and [0038]).

Therefore, in view of Terrill et al, it would have been obvious to one of ordinary skill at the time the invention was made to access the threshold number from a look-up table in order to obtain and store information in job table (as stated in Terrill et al paragraph [0004]).

9. Claim 23 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno '549 in view of Terrill '646, Anderson '624 and Kemp '160.

Re claim 23: Kuno et al teaches a method comprising:

generating a content type that identifies output that is compatible with a printer identified by the printer identification (i.e. the server then makes PDL data using the user's print request that is suitable for the registered printer designated by the user when informing the server (102) about the printer type. The content type that identifies or represents an output that is compatible to the printer is the PDL data that is generated by the server (102); see fig. 1, 5 and 6; paragraphs [0036] and [0037]);

receiving a request to map a printer icon name to a content type (i.e. when the user requests a print, the server receives the request to link the printer icon name, that represents a printer driver, to the content type of the printer designated for printing, or

printer type. The system clearly performs the above feature; see figs. 5 and 6; paragraphs [0036] and [0037]);

mapping a printer icon name to the content type (i.e. the printer name is related, or mapped, to the printer driver that will output a specific content type to the local printer on the client PC. Therefore, the printer name, which represents a printer driver, is mapped or associated with a content type that will be outputted to the local printer; see figs. 5 and 6; paragraphs [0036] and [0037]);

associating a printer driver with the printer icon name (i.e. when it is determined that the server does not have a certain printer driver, it can search for the printer driver by a URL. The URL will return the appropriate printer driver, or some data like a name that represents a printer driver, to the server (102) in order to process a certain content type. The printer driver returned is associated with the printer name that is designated by the user; see figs. 1, 5 and 6; paragraphs [0029], [0036] and [0037]);

determining that the printer driver is available and determining that the printer driver is not available (i.e. in the system, the invention has the ability to allow the user to obtain a printer driver if it is not currently on the server. In order to know that a driver needs to be installed on the server, the determination that a printer driver is or is not on the server has to be made. Therefore, it is clear that the system determines if a printer driver is or is not available. If the printer driver is available, then it can be immediately used and installation of a printer driver is not necessary, which is stated in paragraph [0037]. If it is determined that the printer driver is not available, the server obtains a

Art Unit: 2625

printer driver suitable for outputting the content type; see fig. 5 and 6; paragraphs [0036] and [0037]).

returning the printer icon name if the printer driver is available (i.e. when the CPU (110) checks the hard disk drive to see if a certain printer driver is available, the hard disk drive (116) returns to the CPU (110) the printer driver locator, analogous to a printer icon name, related to a certain printer. At this point, the printer driver may not necessarily need to be reinstalled. Also, when the server (102) checks a URL for a certain printer driver, the URL returns a printer driver in relation to the URL being searched; see figs. 1, 5 and 6; paragraphs [0036] and [0037]).

However, Kuno et al fails to teach comparing a threshold number of print requests to a current number of print requests, if the current number is less than the threshold number and if the threshold number does not exceed the current number.

However, this is well known in the art as evidenced by Terrill et al. Terrill et al discloses

comparing a threshold number of print requests to a current number of print requests (i.e. the system discloses the threshold of a number of print jobs or print jobs completed. The system compares the pre or post print information and the print job completion data to constraints in the system. If constraints in the system are reached, data is transferred to a management server; see paragraphs [0011], [0038] and [0054]);

if the current number is less than the threshold number (i.e. when comparing the print job completion information, post and pre print information with the threshold, the information is compared to see if the threshold is reached. If the threshold is not

reached, it represents the information being compared to the threshold is lower than the threshold; see paragraphs [0011], [0038] and [0054]); and

if the threshold number does not exceed the current number (i.e. when the print job information attains the threshold of the system, then information from the data store is transferred to the management server. This is an example of the threshold not exceeding the current number or the print job number, but being equal to that number; see paragraphs [0011], [0038] and [0054]).

Therefore, in view of Terrill et al, it would have been obvious to one of ordinary skill at the time the invention was made to compare a threshold number of print requests to a current number of print requests, to judge if the current number is less than the threshold number or if the threshold number does not exceed the current number in order to judge to realization of the threshold in the system (as stated in Terrill et al paragraph [0011]).

However, Kuno '549 in view of Terrill '646 fails to teach receiving from a mobile computing device, print instructions and an email having an attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed.

However, this is well known in the art as evidenced by Anderson '624. Anderson '624 discloses receiving from a mobile computing device (i.e. the element 183, which includes a PDA, pager, cell phone or laptop; see fig. 1; paragraph [0026]), print instructions and an email having mail attachment (i.e. the mobile device sends an email with an attachment of documents. The email with the attachment are sent, reformatted,

Art Unit: 2625

to the MPC (106) with special printer requirements that the user desires to be reflected in the output of the attachment; see paragraphs [0035]-[0044]),

the print instructions including a printer identification and document options indicating how the attachment should be printed (i.e. in the information determining the printing of the email attachment, the mobile device sends a printer connector identifier (179) that is associated with the MPC (106), which informs the MPC server (143) which is the destination device to perform the printing of the rendered email attachment. Also, other special printing requirements and settings for the printer that indicates how the user wants the attachment to be printed is also included in the information; see paragraphs [0035]-[0044]).

Therefore, in view of Anderson '624, it would have been obvious to one of ordinary skill at the time the invention was made to have the features of receiving from a mobile computing device, print instructions and an email having ml attachment, the print instructions including a printer identification and document finishing options indicating how the attachment should be printed in order to receive an attachment recognizable by the printer in the mobile printer connector and printing the attachment (as stated in Anderson '624 paragraphs [0007]-[0010]).

However, Kuno '549 in view of Terrill '646 and Anderson '624 fails to teach finishing options.

However, this is well known in the art as evidenced by Kemp '160. Kemp '160 discloses finishing options (i.e. in Kemp '160, the user sends finishing request with the printing request to the service provider (2) from the computer workstation. This feature

Art Unit: 2625

combined with the feature of Anderson '624 to provide options to print the document in a special manner by the printer performs the above feature; see fig. 1; paragraph [0041]).

Therefore, in view of Kemp '160, it would have been obvious to one of ordinary skill at the time the invention was made to have finishing options in order to have the user be able to submit a printer request involving a finishing process (as stated in Kemp '160 paragraph [0041]).

10. Claim 24 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno et al, modified by Terrill et al, Anderson '624, Kemp '160 and Liang '297, and further in view of Chang et al.

Re claim 24: The teachings of Kuno et al in view of Terrill et al are disclosed above.

However, Kuno et al in view of Terrill et al, Anderson '624 and Kemp '160 fails to teach a method, further comprising returning an error message if the printer driver is not available, the error message selected from the group comprising: a message indicating that the printer driver is unavailable; and a message indicating that the printer driver is temporarily unavailable.

Liang '297 teaches a processor-readable medium, wherein the error message is selected from the group comprising:

a message indicating that the printer driver is unavailable (i.e. if a conventional printer driver is not available, an error message is generated after the determination of a

Art Unit: 2625

printer driver is not available. This message indicates that a printer driver is not available; see fig. 8 and 9; col. 7, lines 4-59); and

a message indicating that the printer driver is temporarily unavailable (i.e. if a conventional printer driver is not available, an error message is generated after the determination of a printer driver is not available. This message indicates that a printer driver is not available. Whether temporarily unavailable or unavailable, under the same condition, the printer driver is unavailable; see fig. 8 and 9; col. 7, lines 4-59).

However, Kuno et al, modified by Terrill et al, and further in view of Liang '297 fails to teach that the request should be attempted later.

However, this is well known in the art as evidenced by Chang et al. Chang et al discloses that the request should be attempted later (i.e. the system notifies the user to try again or it notifies that user at a later time for the use of the printer, through the printer driver, when the actual requested service is available; see fig. 5; paragraph [0148]).

Therefore, in view of Chang et al, it would have been obvious to one of ordinary skill at the time the invention was made to have a message indicating that the request should be attempted later in order to notify the user when services become available (as stated in Chang et al paragraph [0148]).

11. Claim 36 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kuno '549 in view of Terrill '646.

Art Unit: 2625

Re claim 36: Kuno et al discloses a print system including application server comprising:

print requests for the printer driver (i.e. in the system for Kuno '549, the client computer communicates with a server in order to obtain a printer driver program to process a print request of a user in the system; see paragraphs [0011], [0030]-[0038])

print requests being processed by the printer driver (i.e. in the system of Kuno, the print requests are handled by a printer driver in order for the print job to be sent to the printer through a suitable printer driver in the system related to the associated printer; see paragraphs [0011], [0030]-[0038]);

means for determining that the printer driver is available and means for determining that the printer driver is not available (i.e. in the system, the invention has the ability to allow the user to obtain a printer driver if it is not currently on the server. In order to know that a driver needs to be installed on the server, the determination that a printer driver is or is not on the server has to be made. Therefore, it is clear that the system determines if a printer driver is or is not available. If the printer driver is available, then it can be immediately used and installation of a printer driver is not necessary, which is stated in paragraph [0037]. If it is determined that the printer driver is not available, the server obtains a printer driver suitable for outputting the content type; see fig. 5 and 6; paragraphs [0036] and [0037]).

However, Kuno et al fails to teach a means for comparing a threshold number of print requests to a current number of print requests being processed, if the current number is less than the threshold number and if the threshold number does not exceed the current number.

However, this is well known in the art as evidenced by Terrill et al. Terrill et al discloses

means for comparing a threshold number of print requests to a current number of print requests being processed (i.e. the system discloses the threshold of a number of print jobs or print jobs completed. The system compares the pre or post print information and the print job completion data to constraints in the system. If constraints in the system are reached, data is transferred to a management server. While the print job information is being processed for storage, the system checks if the information of the print jobs reaches a storage threshold amount of print job information; see paragraphs [0011], [0038] and [0054]);

if the current number is less than the threshold number (i.e. when comparing the print job completion information, post and pre print information with the threshold, the information is compared to see if the threshold is reached. If the threshold is not reached, it represents the information being compared to the threshold is lower than the threshold; see paragraphs [0011], [0038] and [0054]); and

if the threshold number does not exceed the current number (i.e. when the print job information attains the threshold of the system, then information from the data store is transferred to the management server. This is an example of the threshold not exceeding the current number or the print job number, but being equal to that number; see paragraphs [0011], [0038] and [0054]).

Therefore, in view of Terrill et al, it would have been obvious to one of ordinary skill at the time the invention was made to have a means for comparing a threshold

Art Unit: 2625

number of print requests to a current number of print requests, to judge if the current number is less than the threshold number or if the threshold number does not exceed the current number in order to judge to realization of the threshold in the system (as stated in Terrill et al paragraph [0011]).

Conclusion

12. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

13. Nelson '384 (US Pub No 2003/0128384) discloses a system using the servers as the mechanism to drive the printers and the servers compare the amount of print jobs being processed versus the amount of print jobs being requested to be processed by the servers.

14. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of


Art Unit: 2625


the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chad Dickerson whose telephone number is (571)-270-1351. The examiner can normally be reached on Mon. thru Thur. 9:00-6:30 Fri. 9:00-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Aung Moe can be reached on (571)- 272-7314. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

CD/ 
Chad Dickerson
November 19, 2007


AUNG S. MOE
SUPERVISORY PATENT EXAMINER
11/19/07